Erschienen in: Breindl, Eva/Gunkel, Lutz/Strecker, Bruno (Hrsg.): Grammatische Untersuchungen, Analysen und Reflexionen. Festschrift für Gisela Zifonun. - Tübingen: Narr, 2006. S. 601-616. (Studien zur Deutschen Sprache 36)

Roman Schneider

Texttechnologie und Grammatik

1. Einleitung

Sprachwissenschaftliche Kompetenzen und Methoden haben im Verlauf der letzten Jahre erfreulicherweise Zug um Zug Eingang in die praktische wissenschaftliche Arbeit benachbarter Disziplinen gefunden. Nicht zuletzt als Folge der zunehmenden Verbreitung digitaler Medien entstanden darüber hinaus neue, oft interdisziplinär ausgerichtete Forschungsschwerpunkte.

Die Texttechnologie teilt sich mit der allgemeinen Sprachwissenschaft den gemeinsamen Untersuchungsgegenstand, nämlich die Beschäftigung mit natürlicher Sprache. Allerdings konzentriert sie sich dabei primär auf deren elektronische Manifestation in Form digitaler Texte. Das Erkenntnisinteresse betrifft gleichermaßen Syntax, Bedeutung und Verwendung: Wie lassen sich Textstrukturen und Referenzbeziehungen mit Hilfe spezieller Grammatiken modellieren, welche Repräsentationsmodelle unterstützen eine effiziente Informationsextraktion usw. Entsprechend breit gefächert sind die Anknüpfungspunkte an andere linguistische Teildisziplinen. Um einen grundlegenden Eindruck davon zu vermitteln, ob – und wenn ja: wo – wechselseitig profitable Verbindungen zur Grammatikforschung bestehen, will der vorliegende Beitrag einige zentrale texttechnologische Themen sowie deren Anwendungsrelevanz genauer ausleuchten.

2. Markup-Sprachen

Markup-Sprachen (auch: Annotations- oder Auszeichnungssprachen) spielen seit längerem eine wichtige Rolle bei der Planung und Durchführung zahlreicher sprachwissenschaftlicher Forschungsvorhaben. Als prototypische Anwendungsbereiche lassen sich die Erschließung von Korpora, die Erstellung von Wörterbüchern, oder auch die Modellierung von Wort- und Wissensnetzen benennen. In praxi können alle Projekte vom Einsatz adäquater Auszeichnungsmechanismen profitieren, die sich mit der Erfassung, Analyse und algorithmischen Weiterverarbeitung digitalisierter Sprache beschäftigen.

Definition: Das Prinzip der **Markup-Technologie** beruht auf einer formal standardisierten Einbettung unterschiedlichster Meta-Informationen in digitale Dokumente. Vokabular und Syntax von Markup-Sprachen werden mit Hilfe kontextfreier Grammatiken festgelegt. Ziel ist eine eindeutige Identifizierung und Charakterisierung der Dokumentsegmente, idealerweise unabhängig von nachfolgenden Verarbeitungsschritten.

In der Realität lässt sich dieser hohe Anspruch nicht immer konsequent umsetzen, denn jede Annotation erfolgt naturgemäß stets in Abhängigkeit von der jeweils eingenommenen Perspektive und den Zielen, die sich der Bearbeiter gestellt hat. Um die grundlegenden Arbeitsschritte transparenter zu machen, sollte deshalb zwischen einer Annotations- und einer Beschreibungsebene unterschieden werden: Während auf der Annotationsebene die konkrete Auszeichnung vollzogen wird, muss vorher auf der Beschreibungsebene entschieden werden, aus welcher Perspektive und mit welchen Ansprüchen die Analyse der Dokumente durchgeführt werden soll. Beispielsweise wird eine morpho-syntaktische Analyse andere Anforderungen an die Auszeichnung stellen als eine Analyse semantischer oder pragmatischer Relationen, eine spätere kontrastive Untersuchung wird andere Basisinformationen benötigen als die Informationsextraktion aus homogenen, einzelsprachlichen Texten. Alle diese Überlegungen und Entscheidungen führen schließlich im optimalen Fall zu einem adäquaten Datenmodell. Für jede Beschreibungsebene lassen sich ein oder mehrere Annotationsformat(e) formulieren; umgekehrt kann ein Annotationsformat grundsätzlich für eine oder mehrere Beschreibungsebenen einsetzbar sein.

Generell lassen sich zwei Typen von Markup-Sprachen unterscheiden: Deskriptive Markup-Sprachen[1] dienen der Repräsentation struktureller Informationen. Sie beschreiben den formalen Aufbau von Dokumenten, getreu der Devise der strikten Trennung von äußerer Form, Struktur und Inhalt – mit den Worten der Bauhaus-Maxime: „form follows function". Prozedurale Markup-Sprachen[2] hingegen legen primär fest, in welcher Form einzelne Inhaltsbestandteile auf der medialen Präsentationsebene, d.h. auf Bildschirmen oder Druckern, dargestellt werden sollen. Analog hierzu werden auch

[1] Beispielsweise LaTeX oder die Angehörigen der SGML- bzw. XML-Familie, zu der XHTML, SVG, VRML, MathML oder auch die am Institut für Deutsche Sprache in Mannheim eingesetzte Markup-Sprache grammisML gehören.

[2] Prominente Beispiele sind TeX, Postscript oder PDF.

die Begriffe „Generic Markup" respektive „Visual Markup" verwendet. Im Folgenden wird ausschließlich von der erstgenannten Kategorie die Rede sein.

Für die Annotierung syntaktischer Strukturen in digitalen Textsammlungen wurde im Verlauf der letzten Jahrzehnte eine Vielzahl verschiedener Ansätze entwickelt. Die beiden bekanntesten Markup-Lösungen sind sicherlich das in SGML beschriebene Regelwerk der Text Encoding Initiative (TEI) sowie der Corpus Encoding Standard (CES) bzw. dessen XML-basierter Nachfolger XCES.[3] Allerdings weist das TEI-Format – neben seiner nicht gerade geringen Komplexität – den gravierenden Nachteil auf, dass keinerlei Regeln für das Einfügen von morpho-syntaktischen bzw. morphologischen Angaben vorgesehen sind. XCES erscheint in dieser Hinsicht erheblich leistungsfähiger und unterstützt sogar den Einsatz unterschiedlicher Annotationsebenen. Derzeit liegt es noch in einer Beta-Version vor und wird dementsprechend spärlich in aktuellen linguistischen Projekten genutzt. Verbreitet sind weiterhin projektspezifische Beschreibungssprachen, beispielsweise der Tübinger TUSNELDA-Standard[4], das TigerXML-Format des Tiger-Korpus[5] oder das Corpus Document Interchange Format (CDIF) des British National Corpus (BNC)[6]. Das Gesamtbild wird abgerundet durch eher tabellarisch organisierte Lösungen, z.B. das NEGRA-Format[7] oder das Tagset der „Penn TreeBank"[8].

Wie die vorstehende grobe Übersicht bereits vermuten lässt, erscheint das Spektrum relevanter Markup-Sprachen für die linguistische Annotation breit gefächert.[9] Es ist daher wenig verwunderlich, dass bereits seit längerem eine Diskussion über den Nutzen eines theorie- und perspektivenunabhängigen Standards geführt wird. Um die dabei auftretenden Problematiken und den

---

[3] Die TEI-Homepage ist unter `http://www.tei-c.org` erreichbar. Zu CES und XCES, die Bestandteile der Richtlinien der Expert Advisory Group on Language Engineering Standards (EAGLES) und beide grundsätzlich auch TEI-konform sind, vgl. `http://www.cs.vassar.edu/CES/` bzw. `http://www.xml-ces.org`.

[4] Vgl. `http://www.sfb441.uni-tuebingen.de/tusnelda.html`.

[5] Vgl. `http://www.ims.uni-stuttgart.de/projekte/TIGER/`.

[6] Vgl. `http://www.natcorp.ox.ac.uk/`.

[7] Vgl. `http://www.coli.uni-sb.de/sfb378/negra-corpus/`.

[8] Das zugehörige Tagset orientiert sich an den Vorarbeiten zum „Brown Corpus of Standard American English"; vgl. `http://www.cis.upenn.edu/~treebank/`.

[9] Vgl. hierzu auch Naumann (2000) und Ule/Hinrichs (2004). Einen guten Online-Überblick bietet z.B. `http://www.ldc.upenn.edu/annotation/`.

potentiellen Beitrag der Linguistik zu diesem Prozess besser abschätzen zu können, soll nachfolgend kurz auf die sprachtheoretische Einordnung von Markup-Sprachen eingegangen werden.

Für die Klassifizierung von Sprachen aus formaler Sicht hat sich in der einschlägigen Forschung die 1956 von Noam Chomsky vorgeschlagene Sprach- und Grammatik-Hierarchie etabliert. Ursprünglich aus dem Verlangen heraus entstanden, natürliche Sprachen mit formalen Mitteln zu beschreiben, ist diese Einteilung seither nicht nur in der Sprachwissenschaft, sondern insbesondere in der theoretischen Informatik äußerst populär geworden. Im Einzelnen unterscheidet die Chomsky-Hierarchie folgende Sprach- bzw. Grammatiktypen:

- unbeschränkte, rekursiv aufzählbare Sprachen (Typ-0-Grammatik)
- kontextsensitive Sprachen (Typ-1-Grammatik)
- kontextfreie Sprachen (Typ-2-Grammatik)
- reguläre Sprachen (Typ-3-Grammatik)

Zwischen diesen Typen besteht ein echtes Teilmengen-Verhältnis, d.h., jede reguläre Grammatik erfüllt auch sämtliche Anforderungen an eine kontextfreie Grammatik, jede kontextfreie Grammatik erfüllt sämtliche Anforderungen an eine kontextsensitive Grammatik usw. Charakteristisch für die Hierarchie sind eine mit der Erhöhung der Typenbezeichnung einhergehende Beschränktheit der Syntaxregeln sowie sinkende maschinelle Implementierungshürden.

Markup-Sprachen auf der Basis von SGML und XML erfordern die Mächtigkeit von kontextfreien Grammatiken; den Anforderungen an reguläre Sprachen genügen sie nicht. Für die Verwaltung der jeweils zugehörigen Grammatikregeln – d.h. für die Festlegung, welche Elemente und Konstrukte innerhalb einer Markup-Sprache zulässig sind – ist eine sogenannte Document Type Definition (DTD) zuständig. Eine solche DTD besteht, analog zu Grammatiken für natürliche Sprachen, aus Symbolen plus Regeln und kann als Quadrupel $G = (V_N, V_T, P, S)$ beschrieben werden mit:

V_N = endliches Alphabet der Nichtterminalsymbole
V_T = endliches Alphabet der Terminalsymbole
P = endliche Menge der Produktionsregeln
S = Startsymbol aus V_N

Für das Verhältnis von V_N und V_T gilt $V_N \cap V_T = \emptyset$ sowie $V_N \cup V_T = V$. Die Produktionsregeln sämtlicher in der Hierarchie beschriebenen Sprachen entsprechen der Form $\alpha \rightarrow \beta$, wobei $\alpha \in V^*V_NV^*$ und $\beta \in V^*$. Für die Regeln in kontextfreien Sprachen gilt einschränkend $\alpha \in V_N$. Exemplarisch lässt sich der Aufbau einer kontextfreien Grammatik auf XML-Basis – also einer DTD – anhand des folgenden Beispiels verdeutlichen:

(1) *Der Autor arbeitet auf einer alten Schreibmaschine.*

Die Konstituentenstruktur dieses Satzes kann mit Hilfe der Baumansicht in Abbildung 1 visualisiert werden:

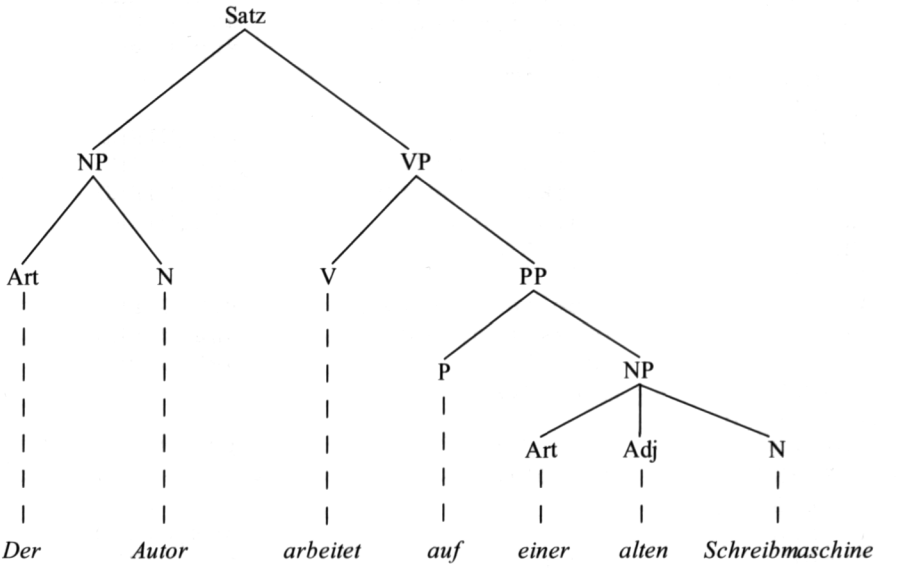

Abb. 1: Baumansicht einer Konstituentenstruktur

Eine zu diesem Satz – und zu allen anderen Sätzen mit exakt der gleichen Struktur – passende DTD käme mit einigen wenigen Syntaxregeln aus, die in ihrer Form an die aus Transformationsgrammatiken bekannten Phrasenstrukturregeln (also z.B. `Satz` → `NP+VP`) erinnern.[10] Um das Beispiel ein wenig anspruchsvoller zu gestalten, sollen jedoch neben der reinen Konstituentenstruktur noch weitere grammatische bzw. morphologische Kategorien einge-

[10] Vgl. z.B. Lobin (2004). Daneben besteht eine Ähnlichkeit zur Backus-Naur-Form (BNF) bzw. Erweiterten Backus-Naur Form (EBNF). Seit ihrer Einführung Anfang der sechziger Jahre werden diese Konventionen für die Syntaxbeschreibung kontextfreier Sprachen (z.B. höherer Programmiersprachen) eingesetzt.

fügt werden. Adjektive, Artikel und Nomen erhalten – teilweise optionale – Attribute für Kasus, Numerus und Genus; Verben erhalten Attribute für Person und Numerus.[11] Artikel sollen definit oder indefinit sein können. Für Präpositionen soll angegeben werden, ob das Folgeelement im Dativ oder im Akkusativ steht:

```
<!ELEMENT satz      (np,vp)>
<!ELEMENT np        (art,adj?,n)>
<!ELEMENT vp        (v,pp)>
<!ELEMENT pp        (p,np)>
<!ELEMENT adj       (#PCDATA)>
<!ATTLIST adj
  kasus             (nom|gen|dat|akk)       #REQUIRED
  genus             (mask|fem|neutr)        #REQUIRED
  numerus           (sg|pl)                 #REQUIRED>
<!ELEMENT art       (#PCDATA)>
<!ATTLIST art
  typ               (def|indef)             #REQUIRED
  kasus             (nom|gen|dat|akk)       #REQUIRED
  genus             (mask|fem|neutr)        #REQUIRED
  numerus           (sg|pl)                 #REQUIRED>
<!ELEMENT n         (#PCDATA)>
<!ATTLIST n
  kasus             (nom|gen|dat|akk)       #IMPLIED
  genus             (mask|fem|neutr)        #REQUIRED
  numerus           (sg|pl)                 #REQUIRED>
<!ELEMENT p         (#PCDATA)>
<!ATTLIST p
  verlangt          (akk|dat)               #REQUIRED>
<!ELEMENT v         (#PCDATA)>
<!ATTLIST v
  person            (1|2|3)                 #REQUIRED
  numerus           (sg|pl)                 #REQUIRED>
```

Eine Auszeichnung des Beispielsatzes (1) gemäß der in dieser XML-DTD eingeführten Elementtypen und Syntaxregeln sähe nun folgendermaßen aus:

```
<satz>
<np>
<art typ="def" kasus="nom" genus="mask"
numerus="sg">Der</art>
<n kasus="nom" genus="mask" numerus="sg">Autor</n>
</np>
<vp>
<v person="3" numerus="sg">arbeitet</v>
<pp>
```

[11] Diese Auswahl deckt natürlich nicht sämtliche charakteristischen Attribute der beschriebenen Wörter ab, sondern ist als exemplarische Auswahl zu sehen.

```
<p verlangt="dat">auf</p>
<np>
<art typ="indef" kasus="dat" genus="fem" nume-
rus="sg">einer</art>
<adj kasus="dat" genus="fem" numerus="sg">alten</adj>
<n genus="fem" numerus="sg">Schreibmaschine</n>
</np>
</pp>
</vp>
</satz>
```

Die Vorzüge einer solchen XML-basierten Kodierung linguistischer Informationen liegen in der Flexibilisierung der weiteren Anwendung. Alle standardisierten Hilfsmittel, die mittlerweile für die XML-Sprachenfamilie verfügbar sind, können unmittelbar genutzt werden, um Inhalte bei jederzeit garantierter Datenintegrität zu bearbeiten. Editoren, Parser und Transformationswerkzeuge erleichtern den Austausch mit anderen Quellen und Anwendungen. Außerdem helfen sie bei der Erweiterung bestehender DTD-Modelle um zusätzliche Elementtypen oder beim Anlegen von Verknüpfungen. Im Übrigen lassen sich Markup-Texte, verglichen mit anderen Repräsentationsformen wie z.B. Tabellen oder Klammerstruktur-Formaten, bis zu einem gewissen Komplexitätslevel nicht nur durch Computer verarbeiten, sondern erschließen sich auch dem menschlichen Betrachter.

Neben der bislang betrachteten integrierten Repräsentation linguistischer Angaben in der XML-Quelle bietet sich in manchen Fällen eine verteilte Repräsentation an. Diese kann bereits dann nützlich erscheinen, wenn die schiere Menge und Schachtelung der Elementtypen sowie die Anreicherung um Attribute die Grenze dessen überschreitet, was mit zumutbarem Aufwand noch von einem menschlichen Nutzer erfasst werden kann. Eine Verteilung der Meta-Informationen auf verschiedene Dateien wird spätestens dann unvermeidlich, wenn verschiedene Modelle oder Perspektiven zu berücksichtigen sind; in diesem Zusammenhang ist oft auch von „Stand-Off Markup" und „Multi-Level Annotation" die Rede. Angenommen, die obige XML-Instanz sollte um die Ergebnisse einer automatischen Lemmatisierung ergänzt werden. Die integrierte Lösung bestünde darin, für die betroffenen Elementtypen ein zusätzliches Attribut zu definieren und dieses dann entsprechend aufzufüllen. Dies lässt sich anhand des Verb-Elements demonstrieren:

```
<v lemma="arbeiten" person="3" numerus="sg">arbeitet</v>
```

Eine verteilte Lösung, wie sie etwa auch XCES unterstützt (vgl. Ide/Romary 2003), würde dagegen zunächst lediglich die rudimentären Strukturinformationen, also z.B. Absatz- und Satzgrenzen, direkt in die Primärdaten einbetten:

```
<satz id="s1">Der Autor arbeitet auf einer alten
Schreibmaschine</satz>
```

Alle weiterführenden Angaben könnten dann mittels XLink-Verknüpfungen und XPointer-Adressen in separaten Dateien vorgehalten werden:[12]

```
<chunk xml:base="http://www.ids-mannheim.de/test.xml#">
<struct
 id="t3"
 xlink:href="xptr(substring(//s[id="s1"]/text(),11,8))">
 <feat type="cat">v</feat>
 <feat type="lemma">arbeiten</feat>
 <feat type="person">3</feat>
 <feat type="numerus">sg</feat>
</struct>
</chunk>
```

Eine derartige Verbindung normierter Markup-Sprachen auf XML-Basis mit den flankierenden Standards zur Verknüpfung und Datenextraktion dürfte zukünftig einen bedeutenden Einfluss auf Projekte haben, die sich mit der Anreicherung digitaler Sprachdaten um linguistische Informationen beschäftigen – insbesondere dann, wenn mehrere verschiedene Perspektiven bzw. Beschreibungsebenen unterschieden werden müssen. Die Primärdaten lassen sich auf diese Weise aufgrund ihrer flachen und einfachen Struktur für verschiedene Zwecke wiederverwenden und ebenso leicht zwischen kooperierenden Forschergruppen austauschen.

3. Wissensorganisation und -repräsentation

Wie bereits in der Einleitung des vorigen Abschnitts angesprochen, spielen Markup-Sprachen unter anderem bei der Organisation und Repräsentation digitalisierten Wissens eine zentrale Rolle. Und in der Tat konstituiert gerade die Auseinandersetzung mit diesem Zusammenwirken einen gleichermaßen innovativen wie ertragreichen Forschungsgegenstand der Texttechnologie. Im Mittelpunkt des Interesses stehen dabei die Standards und Methoden für eine explizite systematische Festschreibung von Entitäten und bedeutungstragenden Relationen eines Fachgebiets mit dem Ziel, dieses geordnete „Wissen" zur Erschließung digitaler Textsammlungen zu nutzen.

---

[12] Zu XLink und XPointer vgl. z.B. Behme/Mintert (2000).

Noch vor wenigen Jahren erschien der Kreis derjenigen, die sich für den Aufbau digitaler Wissensrepräsentationssysteme interessierten, als vergleichsweise übersichtlich: In den Forschungen zur Künstlichen Intelligenz (KI) und zur Verarbeitung natürlicher Sprache beschäftigte man sich mit der Funktion von Sprache als Vermittler zwischen Mensch und Umwelt und modellierte in diesem Zusammenhang beispielsweise unterschiedlich komplexe semantische Netze – zumeist für klar umgrenzte Anwendungsdomänen, gelegentlich auch zur Abbildung fachübergreifenden Weltwissens. Mittlerweile kommen verwandte Systeme und Technologien auf breiterer Basis zum Einsatz. Im Kontext der maschinellen Informationserschließung haben sich insbesondere Thesauri und Ontologien als leistungsfähige Instrumente erwiesen. Da beide Begriffe häufig synonym, gelegentlich sogar in widersprüchlicher Weise verwendet werden, empfiehlt sich zunächst eine kurze definitorische Klarstellung.

Definition: Als **Thesaurus** bezeichnen wir ein Klassifikationssystem, in dem ausgewählte Terme (auch: Ausdrücke oder Deskriptoren) einer Sprache bzw. eines Fachgebiets als kontrolliertes Vokabular dargestellt werden. Zusätzlich erfasst ein Thesaurus hierarchische Relationen (Hyponymie und Hyperonymie) und lexikalische Relationen (Synonymie, Antonymie, verwandte Ausdrücke).

Thesauri sind als Ordnungs- und Recherchehilfsmittel somit mächtiger als einfache Taxonomien, die beispielsweise von Internet-Verzeichnissen wie *Yahoo!* verwendet werden, oder Kataloge für Angebotsplattformen wie *Amazon*. Sowohl Taxonomien wie auch Kataloge erfassen zwar ebenfalls kontrollierte Vokabularien, bilden allerdings bestenfalls einfache, nicht typisierte Hierarchiebeziehungen ab. Thesauri können darüber hinaus zumindest ein begrenztes Inventar aussagekräftigerer Beziehungen kodieren, was wiederum eine Grundvoraussetzung für ihren Einsatz im Rahmen einer anspruchsvollen automatischen Informationserschließung ist.[13] Exemplarisch für diesen Einsatzbereich stehen Fragestellungen, die mit der Auflösung von Synonymie- und Polysemieproblemen verbunden sind.

Strukturierungsvorschläge für Thesauri liegen mittlerweile in Form einschlägiger Richtlinien vor. ISO-2788:1986 und ANSI Z39.19-1993 sowie das

[13] Daneben werden Thesauri auch als stilistische Hilfsmittel in modernen Textverarbeitungsprogrammen eingesetzt oder dienen der manuellen Erschließung von Sachgebieten wie z.B. der deutschsprachige „OpenThesaurus“ unter `http://www.openthesaurus.de`.

deutsche Pendant DIN-1463-1 definieren Benennung, Abkürzung und Verwendung der möglichen Relationen für monolinguale Thesauri; entsprechende Vorgaben existieren für multilinguale Systeme. Darauf aufbauend wurden in der Vergangenheit eine Reihe von Lösungsvorschlägen gemacht, die eine normierte Implementierung von Thesauri unterstützen. Neben RDF-basierten Ansätzen[14] erscheinen in diesem Zusammenhang hauptsächlich XML-Sprachen interessant, die neben einer konsequenten Umsetzung der Grundlagen auch Erweiterungen zur Erfüllung individueller Projektbedürfnisse erlauben. Exemplarisch können hier die Thesaural Markup Language (TML), die Schemata des Open University Thesaurus oder die Zthes-Spezifikation genannt werden.[15] Letzte soll im nachfolgenden Beispiel genutzt werden, um den möglichen Aufbau eines sprachwissenschaftlichen Thesaurus aufzuzeigen:

```
<term>
      <termId>10</termId>
      <termName>Nominalphrase</termName>
      <relation>
            <relationType>BT</relationType>
            <termId>1</termId>
            <termName>Phrase</termName>
      </relation>
      <relation>
            <relationType>RT</relationType>
            <termId>22</termId>
            <termName>Nomen</termName>
      </relation>
      <relation>
            <relationType>UF</relationType>
            <termId>11</termId>
            <termName>Nominalgruppe</termName>
      </relation>
</term>
```

Beschrieben wird hier eine ausgewählte Menge von Beziehungen[16] für den Term `Nominalphrase`. Zum Relationstyp `BT` („Broader Term"), der eine hierarchische Überordnung des Terms `Phrase` ausdrückt, existiert in der

[14] Hierzu zählen beispielsweise die für das zukünftige „Semantic Web" erarbeiteten Entwürfe; vgl. `http://www.w3.org/2001/sw/Europe/reports/thes/rdfthes.html`.

[15] Vgl. Lee/Baillie/Dell'Oro (1999) sowie `http://guardians.open.ac.uk/schemas/thesaurus/` und `http://zthes.z3950.org`.

[16] Die Zthes-konforme Benennung der Relationstypen `BT`, `NT`, `RT`, `UF` und `USE` folgt den Vorgaben von ISO-2788; DIN-Entsprechungen wären `OB` („Oberbegriff"), `UB` („Unterbegriff"), `VB` („Verwandter Begriff"), `BF` („Benutzt für") und `BS` („Benutze Synonym").

Strukturspezifikation das konverse Gegenstück NT („Narrower Term"). Der Relationstyp RT („Related Term") verweist auf den verwandten Term Nomen, wobei Art oder Begründung dieser Verwandtschaft nicht weiter ausgeführt sind. Der Relationstyp UF („Use For") kennzeichnet eine Äquivalenzrelation zum Synonym Nominalgruppe; zur Kennzeichnung einer bevorzugten Äquivalenzrelation könnte USE verwendet werden.[17]

Das Beispiel verdeutlicht eine prinzipielle Beschränkung von Thesauri, nämlich die fehlende Möglichkeit zur Formulierung komplexerer Beziehungen. Während sich der Zusammenhang zwischen Phrase und Nominalphrase mit Hilfe der Allgemeiner-Spezifischer-Relation noch angemessen ausdrücken lässt, ist dies für den Zusammenhang von Nominalphrase und Nomen nicht mehr möglich. Das Nomen ist ja eben nicht „eine Art von Nominalphrase", sondern möglicher – in Einzelfällen sogar alleiniger – Bestandteilteil einer Nominalphrase. An dieser Stelle setzen mächtigere Repräsentationssysteme mit einer höheren Aussagekraft an. Im Folgenden soll für diese Systeme der Begriff „Ontologien" verwendet werden.[18]

Definition: Unter einer **Ontologie** verstehen wir die konsistente formale Beschreibung ausgewählter Konzepte (auch: Klassen) einer Anwendungsdomäne. Charakteristisch ist die Vererbung von Eigenschaften (auch: Attribute oder Slots) von allgemeineren auf speziellere Konzepte. Wechselseitige Beziehungen unterschiedlichster Art werden präzise vermittels explizit benannter Relationstypen beschrieben.

Eine solche Definition schließt übrigens auch lexikalisch-semantische Wortnetze wie das Princeton WordNet oder dessen deutsches Gegenstück

---

[17] Eine Diskussion darüber, wann ein Ausdruck als Synonym eines anderen Ausdrucks anzusehen ist, soll an dieser Stelle nicht geführt werden. Für die Zwecke der maschinellen Informationserschließung dürfte in den meisten Fällen eine allzu enge Auslegung – totale Synonymie bei uneingeschränkter Austauschbarkeit – eher hinderlich sein, und die Austauschbarkeit in einem bestimmten Kontext als ausreichendes Kriterium bevorzugt werden.

[18] Aus wissenschaftstheoretischer Sicht mag dieser Begriff „ein paar Nummern zu groß" (Ferber 2003, S. 59) gewählt sein. In der KI-Forschung – und in der Folge auch in texttechnologischen Anwendungen zur Informationserschließung – wird darunter jedoch nicht die „Lehre vom Sein" verstanden, sondern ganz pragmatisch eine – nicht notwendigerweise vollständige – Sammlung von Fakten, die mit Hilfe eines Computers modelliert werden sollen; vgl. z.B. Guarino (1998).

GermaNet ein.[19] In dieses Netzen übernehmen so genannte „Synsets" – Zusammenfassungen von bedeutungsgleichen Begriffen zu einer elementaren Repräsentationseinheit – die Rolle von Konzepten. Die Aussagekraft von Ontologien wird bei einer Betrachtung der komplexen Beziehungen deutlich, die in Abbildung 2 skizziert sind:

- Relation ①: Der Zusammenhang zwischen `Nominalphrase` und `Phrase` kann, analog zum Vorgehen im obigen Thesaurus-Beispiel, durch eine **Hyponymie-Beziehung** (Teilmengen-Beziehung) ausgedrückt werden.
- Relation ②: `Nomen` und `Nominalphrase` lassen sich dagegen adäquater durch eine **Meronymie-Beziehung** (Teil-Ganzes-Beziehung) verbinden. `Nomen` ist Meronym zu `Nominalphrase` und `Nominalphrase` das Holonym zu `Nomen`.
- Relation ③: Zwischen `Nominalphrase` und `Nominalgruppe` besteht eine **Synonymie-Beziehung**.
- Relation ④: Jede Phrase besitzt einen lexikalischen Kopf. Dies lässt sich durch Zuweisung einer entsprechenden **Eigenschaft** ausdrücken, die an alle Hyponyme vererbt wird.
- Relation ⑤: *`Buch`* ist eine Instanz des Konzepts `Nomen` und mit diesem über eine **Element-Beziehung** verbunden.

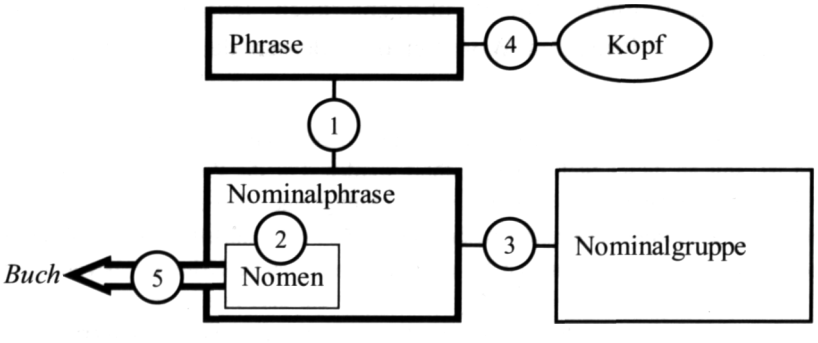

Abb. 2: Modellierung komplexer Relationen

[19] Zu WordNet vgl. z.B. Fellbaum (1998); zu GermaNet vgl. z.B. Kunze (2001).

Der Mehrwert einer solchen Modellierung liegt insbesondere in der Präzisierung der Allgemeiner-Spezifischer-Relation mit Hilfe der Hyponymie-, Meronymie- und Element-Beziehungen. Auch das Anlegen von Kreuzklassifikationen, d.h. die Unterordnung eines Konzepts unter verschiedene übergeordnete Konzepte, ist möglich, da sich das Modellierungsinventar nicht auf monohierarchische Strukturen beschränkt.

Zwar kann das Ziel jeder Modellierung nur die Abstraktion (sprich: Vereinfachung) und keine umfassende Erklärung eines Fachbereichs sein. Trotzdem erlaubt eine konsistente Konzeptionalisierung prinzipiell auch die Anbindung an Inferenzsysteme und damit die Ableitung impliziten Wissens. Voraussetzung hierfür ist die Beschreibung von Ontologien mit Hilfe ausdrucksstarker Auszeichnungssprachen. In den letzten Jahren hat in diesem Bereich, neben konkurrierenden Standards wie XML Topic Maps (XTM), der XML-Based Ontology Exchange Language (XOL) oder DAML+OIL,[20] die Web Ontology Language (OWL) von sich reden gemacht. OWL gehört zur Familie der Beschreibungslogiken, verwendet das Vokabular von RDFS (RDF Schema) und liegt in den drei unterschiedlich expressiven Varianten OWL Lite, OWL DL (Description Logics) und OWL Full vor.[21]

Um einen Eindruck der Funktionsweise von OWL zu vermitteln, sollen abschließend zwei kurze und prägnante Beispiele folgen. Relation ① – also `Nominalphrase ⊆ Phrase` – ließe sich folgendermaßen umsetzen:

```
<owl:Class about = "#Nominalphrase">
  <rdfs:subClassOf rdf:resource = "#Phrase" />
</owl:Class>
```

Die Eigenschaft, dass jede Phrase genau einen lexikalischen Kopf besitzt (Relation ④), lässt sich in OWL durch Formulierung einer entsprechenden Restriktion ausdrücken. Ergänzend dürften an anderer Stelle weitere Relationen festgelegt werden, etwa dass sich dieser Kopf aus dem Inventar einer feststehenden Wortklassenliste rekrutieren muss.

---

[20] Vgl. `http://www.topicmaps.org`, `http://xml.coverpages.org/xol.hml`, `http://www.daml.org`.

[21] Die Empfehlung des World Wide Web Consortiums zu OWL findet sich unter `http://www.w3.org/TR/owl-features/`. Im Zusammenhang mit dem „Semantic Web" vgl. auch `http://www.semanticweb.org` sowie Berners-Lee/Hendler/Lassila (2001).

```
<owl:Class about = "#Phrase">
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource = #hatKopf />
      <owl:cardinality>1</owl:cardinality>
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>
```

4. Anwendungen und Ausblick

Im Rahmen des vorliegenden Beitrags wurde versucht, verschiedene Ebenen aufzuzeigen, auf denen es einen fruchtbaren Austausch von Methoden und Erkenntnissen zwischen Texttechnologie und Grammatik geben kann. Einerseits profitiert die Grammatikforschung vom Einsatz texttechnologischer Verfahren: Der Einsatz von Markup-Sprachen bei der Anreicherung digitaler Textsammlungen um linguistische Meta-Informationen oder bei der Strukturierung grammatischer Wörterbücher erhöht deren praktischen Nutzwert, erleichtert die statistische Auswertung von Korrelationen und ermöglicht dadurch die Evaluierung von Theorien. Für das Anwendungsgebiet der automatischen Informationsverarbeitung ließe sich gegebenenfalls untersuchen, ob Inhalte aus bestimmten syntaktischen Einheiten (z.B. Nominalphrasen) einen anderen informativen Stellenwert besitzen als die Inhalte sonstiger Wortgruppen. Elektronische Thesauri und Ontologien befördern darüber hinaus die terminologische Konsistenz und Erschließung von Fachpublikationen.[22]

Andererseits bietet sich der Linguistik die Chance, eigene Erfahrungen und Kompetenzen in die Entwicklung anwendungsrelevanter texttechnologischer Standards einfließen zu lassen. Exemplarisch kann hier der Umgang mit diskontinuierlichen Strukturen bei der Erstellung von Markup-Grammatiken genannt werden. Die Problematik überlappender Elemente, aus Grammatik und Intonation hinreichend bekannt, zählt zu den Hauptschwierigkeiten beim Einsatz von Markup-Sprachen. Interessant erscheint weiterhin die Frage, ob sich der enorme intellektuelle Aufwand für die Formulierung von DTDs durch eine automatisierte Herleitung von Strukturregeln aus Textkorpora reduzieren ließe.

---

[22] In diesem Zusammenhang haben z.B. Herbermann/Gröschel/Waßner (2002) bereits wertvolle Vorarbeiten geleistet.

Ein erfolgreiches Beispiel für das Zusammenspiel von Texttechnologie und Grammatikforschung stellt das am Institut für Deutsche Sprache (IDS) in Mannheim beheimatete grammatische Informationssystem *grammis* dar.[23] In Anknüpfung an die in Printform vorliegende „Grammatik der Deutschen Sprache“[24] wurde hier zunächst an einer hypermedialen Umsetzung und Erweiterung gearbeitet. Dabei galt es, den klassischen Spagat zwischen Analyseaufwand und Trennschärfe zu bewältigen: Die Repräsentation, d.h. die Grammatik der maßgeschneiderten Markup-Sprache grammisML, sollte mächtig genug sein, sämtliche als relevant erachteten Strukturen ausdrücken zu können – und gleichzeitig so einfach wie möglich, um eine manuelle Annotation und automatische Analyse nicht unnötig zu erschweren. Später rückten weitere texttechnologische Aspekte ins Blickfeld: Die in der Wissensbank hinterlegten Informationen wurden mit Hilfe eines fachgebietsspezifischen Thesaurus umfassend erschlossen, das Auffinden von flektierten Wortformen und Phrasen ermöglicht, und die Recherche-Komponente um einen Erkennungsmechanismus für typische Tippfehler bereichert.

5. Literatur

Behme, Henning/Mintert, Stefan (2000): XML in der Praxis. Professionelles Web-Publishing mit der Extensible Markup Language. Bonn: Addison-Wesley.

Berners-Lee, Tim/Hendler, James/Lassila, Ora (2001): The Semantic Web. In: Scientific American. May 2001, S. 23-43.

Fellbaum, Christiane (Hg.) (1998): WordNet – An Electronical Lexical Database. Language, Speech, and Communication. Cambridge, MA/London: MIT Press.

Ferber, Reginald (2003): Information Retrieval. Suchmodelle und Data-Mining-Verfahren für Textsammlungen und das Web. Heidelberg: dpunkt.verlag.

Guarino, Nicola (1998). Formal Ontology and Information Systems. In: Guarino, Nicola (Hg.): Formal Ontology in Information Systems. Amsterdam: IOS Press. S. 3-15.

Herbermann, Clemens-Peter/Gröschel, Bernhard/Waßner, Ulrich Hermann (2002): Sprache & Sprachen 2. Thesaurus zur Allgemeinen Sprachwissenschaft und Sprachenthesaurus. Wiesbaden: Harrassowitz.

Ide, Nancy/Romary, Laurent (2003): Encoding Syntactic Annotation. In: Abeillé, Anne (Hg.): Treebanks: Building and Using Parsed Corpora. Dordrecht: Kluwer. S. 281-296.

---

[23] Vgl. Schneider (2004).

[24] Vgl. Zifonun/Hoffmann/Strecker et al. (1997).

Kunze, Claudia (2001): Lexikalisch-semantische Wortnetze. In: Carstensen, Kai-Uwe et al. (Hg.): Computerlinguistik und Sprachtechnologie. Heidelberg/Berlin: Spektrum Akademischer Verlag. S. 386-393.

Lee, Maria/Baillie, Steward/Dell'Oro, Jon (1999): TML: A Thesaural Markup Language. In: Proceedings of the 4th Australasian Document Computing Symposium, Coffs Harbour, Australia, December 3, 1999. S. 15-22.

Lemnitzer, Lothar/Lobin, Henning (Hg.) (2004): Texttechnologie. Perspektiven und Anwendungen. Tübingen: Stauffenburg.

Lobin, Henning (2004): Textauszeichnung und Dokumentgrammatiken. In: Lemnitzer/Lobin (Hg.), S. 51-82.

Naumann, Sven (2000): XML als Beschreibungssprache syntaktisch annotierter Korpora. In: Seewald-Heeg, Uta (Hg.): Sprachtechnologie für die multilinguale Kommunikation – Textproduktion, Recherche, Übersetzung. Sankt Augustin: Gardez! Verlag. S. 376-390.

Schneider, Roman (2004): Benutzeradaptive Systeme im Internet: Informieren und Lernen mit GRAMMIS und ProGr@mm. Mannheim: IDS. (= amades 4/04).

Ule, Tylman/Hinrichs, Erhard (2004): Linguistische Annotation. In: Lemnitzer/Lobin (Hg.), S. 195-216.

Zifonun, Gisela/Hoffmann, Ludger/Strecker, Bruno et al. (1997): Grammatik der deutschen Sprache. 3 Bde. Berlin/New York: de Gruyter. (= Schriften des Instituts für deutsche Sprache 7.1-7.3).